UMA telecon 2020-08-20

UMA telecon 2020-08-20

Date and Time

- **Alternate-week Thursdays 6:30am PT**
  - Screenshare and dial-in: https://global.gotomeeting.com/join/485071053
  - United States: +1 (224) 501-3316, Access Code: 485-071-053
  - See UMA calendar for additional details: http://kantarainitiative.org/confluence/display/uma/Calendar

Agenda

- Approve minutes of UMA telecon 2020-07-09, 2020-07-16, 2020-07-23, 2020-07-30, 2020-08-06, 2020-08-13
- Relationship Manager profile
- AOB

Minutes

Roll call

Quorum was not reached.

Approve minutes

- Approve minutes of UMA telecon 2020-07-09, 2020-07-16, 2020-07-23, 2020-07-30, 2020-08-06, 2020-08-13

Deferred.

PKCE and UMA; why to identify clients; who has a stake in identifying clients

Discussion with George: As long as the permission ticket is refreshed on every claims gathering round (which it is), and only the most recent permission ticket is accepted in the next round (which it should be), we should be okay. PKCE is trying to prevent "waking up someone else's app". Android doesn't have quite the same discipline as Apple around scheme/app checking. Our analogy is permission ticket .eq. authorization code. If the code challenge is sent in an ICG flow, does it have to be sent to the token endpoint? We might need to provide some direction around what to do if the client generates the new value, or possibly say it shouldn't. Is getting the same hash multiple times better than multiple hashes in multiple rounds? Interception of the response would be the problem. We could write a profile of DPOP. The client could pass in its key, which would get bound to the permission ticket. That would prevent any intercepted message from an alternative client from being interpreted as legitimate. We could also be more prescriptive around DCR. DPOP was meant to help in the situation around SPAs. For mobile apps, you register it with the AS, and then you can use whatever POP method you want. OAuth 2.1 helps a bit (as previously noted) because PKCE is mandated; this brings client identification. DCR brings a persistent client identification and authentication mechanism.

AI: Alec: Continue to massage the UIG blurb to include these new insights.

In a bring-your-own-device context (which we now frequently see in healthcare and other sectors), we're finding it important to dynamically register (identify and subsequently authenticate) UMA client applications used with those devices. Adrian notes that there is some trouble out there handling software statements for clients. And he's aware of US law that says that a client application unrecognized by an AS can't be refused, and its user, the patient, can only be issued a "black box warning" (there is no distinction in the law between blocking the client and blocking the patient). George believes UMA fits in this model quite a bit better than OAuth because it doesn't require "trust" in the client. Then a process begins to look at proof around the RqP entity against the policies protecting the resource.

There is little awareness among people not at this table that RS's are able to outsource client management to the AS simply because the client can remain relatively untrusted; as long as the client is simply persistently identifiable throughout a single run of the protocol (it's "the same dog" throughout), it shouldn't much matter what it actually is, because claims would be gathered about the RqP to satisfy policy in granting resource access. Even though this feature is native to UMA, the resource definition profile seems to enhance the comfort level of RS's in making the AS a kind of community trust anchor for clients (not the original purpose of the profile; it just worked out this way). Could we complete a deeper analysis of the native feature and then publicize it better?

Leveraging RAR for the client presenting claims could be interesting, and PAR could be used as a security technique as well.

Adrian mentioned the work of IEEE's "P2933 - Clinical IoT Data and Device Interoperability with TIPPSS (EMB/Stds Com/Clinical IoT DDI... Home".

George mentioned: "Google's WebID proposal is in the W3C Incubator Community Group (WICG) and Apple's isLoggedIn() proposal is being discussed in the W3C Privacy Community Group (WPCG)" If people can get involved in these communities, that's helpful. Not directly related to UMA, but browser support for identity stuff is relevant.

Lots of stuff we could profile here, in order to take advantage of new OAuth-related tools to strengthen UMA's abilities.

Relationship Manager profile

Alec's latest spec text is in this thread.

Deferred this time.

Attendees

As of July 8, 2020, quorum is 6 of 10. (Michael, Domenico, Peter, Sal, Gaurav, Thomas, Andi, Maciej, Eve, Mike)

1. Michael
2. Sal
3. Thomas
4. Eve

Non-voting participants:

- George
- Scott
- Adrian
- Anik
- Alec
- Colin

Regrets:

- Domenico